

利用擦痕对节理配套的方法及数据自动处理 Mating of Joints and Data Automatically Processing with Slickensides

张明华
Zhang Minghua

王增科
Wang Zengke

(桂林工学院资源与环境工程系 桂林 541004)
(Dept. of Resource and Env. Eng., Guilin
Institute of Tech., Guilin, 541004)

(江西德兴铜矿 江西德兴 334200)
(Dexing Copper Ore of Jiangxi,
Dexing, Jiangxi, 334200)

摘要 根据剪节理面上的擦痕与主应力轴的空间几何关系,由所测得的剪节理组的产状、剪节理面上的擦痕的方位,用定量求解的方法进行节理配套,找出与之配套的另一组剪节理。用计算机 C 语言设计出程序,借助微机自动完成其复杂的分析处理及配套过程。

关键词 擦痕 剪节理 C 程序

中图法分类号 O29

Abstract According to the geometric relations between main stress axes and slickensides on the plane of shear joints, joints can be matched quantitatively with attitudes of shear joint groups and orientations of slickensides on the plane of shear joint. The another joint group matching the current group can be found. The complicated analysis and match are completed automatically by means of microcomputer with C program.

Key words slickenside, shear joint, C program

在传统的地质工作中,擦痕常用来定性地判断断层两盘的相对运动方向,除此之外,擦痕也可以用于对共轭剪节理配套。节理研究中,多用互切互错、追踪、尾端变化等判定节理的配套关系^[1],但在节理期次较多、节理很发育的地区,根据上述特征判定共轭关系往往比较困难,在缺乏明显判别标志情况下,我们也可以用剪节理面上的擦痕定量计算、研究其配套问题。

1 基本原理

在同一期应力场中形成的共轭剪节理具有特定的运动学性质,即两组共轭剪节理的交线为中间应力轴(σ_2)方位,节理面两侧岩块错移方向(即擦痕)和中间应力轴(σ_2)同位于剪节理

面上且相互垂直(图1)。根据这一基本原理,若测得节理面上擦痕的侧伏向、侧伏角,就可以对节理配套。

2 算法

2.1 划分节理组

对观察点的诸多节理划分节理组,测量或求出优势方位,确定有擦痕节理所在的节理组。

设两面状构造的产状为 $\Phi_1/\theta_1, \Phi_2/\theta_2$ (Φ_1, Φ_2 为倾向, θ_1, θ_2 为倾角), 则其夹角 α_p 为:

$$\alpha_p = \arccos(\sin\theta_1\sin\theta_2\cos(\Phi_2 - \Phi_1) + \cos\theta_1\cos\theta_2).$$

在设定的误差范围(小于 10°) 内,由夹角的大小确定所属节理组。

2.2 求中间应力轴(σ_2)

擦痕与中间应力轴 σ_2 存在一定的几何关系:侧伏向相反,侧伏角互余。设剪节理产状为 Φ_i/θ_i (Φ_i 为倾向, θ_i 为倾角),节理面上擦痕侧伏角为 β_i ,中间应力轴 σ_2 的产状为 $\omega_{\sigma_2}/\delta_{\sigma_2}$ (ω_{σ_2} 为倾伏向, δ_{σ_2} 为倾伏角), 则:

$$\delta_{\sigma_2} = \arcsin(\sin\beta_i\sin\theta_i),$$

$$\omega_{\sigma_2} = \Phi_i - \arcsin(\cos\beta_i/\cos\delta_{\sigma_2}).$$

2.3 节理配套

求其他节理组与带擦痕节理组交线的产状,比较这些交线产状与 σ_2 方位,与 σ_2 最接近者(理论上应相同)即为配套的共轭剪节理组。

若两面状构造产状为 $\Phi_1/\theta_1, \Phi_2/\theta_2$ (Φ_1, Φ_2 为倾向, θ_1, θ_2 为倾角),交线产状为 ω_j/σ_j (ω_j 为倾伏向, σ_j 为倾伏角), 则:

$$\omega_j = \arctan(y/x),$$

$$\sigma_j = \arccos(z/(x^2 + y^2 + z^2)^{1/2}),$$

其中 (x, y, z) 为其交线的空间坐标,其值为:

$$x = -\sin\theta_1\sin\Phi_1\cos\theta_2 + \sin\theta_2\sin\Phi_2\cos\theta_1,$$

$$y = \sin\theta_1\cos\Phi_1\cos\theta_2 - \sin\theta_2\cos\Phi_2\cos\theta_1,$$

$$z = \sin\theta_1\sin\theta_2\sin(\Phi_2 - \Phi_1).$$

设两线状构造的产状为 $\omega_1/\delta_1, \omega_2/\delta_2$ (ω_1, ω_2 为倾伏向, δ_1, δ_2 为倾伏角), 则其夹角 α_l 为:

$$\alpha_l = \arccos(\cos\delta_1\cos\delta_2\cos(\omega_2 - \omega_1) + \sin\delta_1\sin\delta_2).$$

交线产状与 σ_2 夹角在设定的误差范围(小于 10°) 内的节理组即为与之共轭的另一节理组。

3 擦痕数据的微机处理

为提高工作效率和运算精确度,擦痕数据的处理可以借助微机用程序来完成。在前人研究^[2]基础上,我们根据现今微机的软硬件配置,用流行的、移植性好的 C 语言对此重新进行程

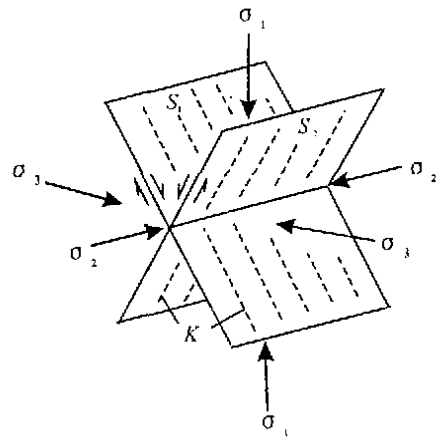


图1 共轭剪节理、擦痕、主应力方位空间关系
 S_1, S_2 - 共轭剪节理; K - 擦痕; σ_1 - 最大主应力; σ_2 - 中间主应力; σ_3 - 最小主应力

序设计。

3.1 程序流程

根据上述基本原理及数学算法,设计程序流程如图 2。

3.2 C 程序设计

程序总体上为顺序结构,根据需要,程序中自定义了 2 个大类共 9 个函数。

(1) 计算类:

$f2(a1, b1, a2, b2)$ 求面与面之间夹角, $a1, b1$ 和 $a2, b2$ 分别为 2 个面状构造的倾向、倾角。

$f3(a1, b1, a2, b2)$ 求线与线之间夹角, $a1, b1$ 和 $a2, b2$ 分别为 2 个线状构造倾向、倾角。

$f4_a(a1, b1, a2, b2)$ 求 2 个面状构造交线的倾向, $a1, b1$ 和 $a2, b2$ 分别为 2 个面状构造的倾向、倾角。

$f4_b(a1, b1, a2, b2)$ 求 2 个面状构造交线的倾角, $a1, b1$ 和 $a2, b2$ 分别为 2 个面状构造的倾向、倾角。

$f1_a(a, b, c, d)$ 求擦痕的倾向, a, b 为节理倾向、倾角, c 为擦痕侧伏角, d 为擦痕侧伏向。

$f1_b(b, c)$ 求擦痕的倾角, b 为节理倾角, c 为擦痕侧伏角。

(2) 绘图类:

$drawplane(a, b)$ 绘制面状构造赤平投影, a, b 为面状构造倾向、倾角。

$drawline(a, b)$ 绘制线状构造赤平投影, a, b 为线状构造倾向、倾角。

$wf()$ 绘制吴氏网、无参数。

3.3 C 程序及使用方法

```

/* 节理配套 C 程序 */
#include "graphics.h"
#include "math.h"
#include "stdio.h"
#include "stdlib.h"
#include "conio.h"
#define R 90
#define K (180/3.14159)
float x0=300;
float y0=205;
void drawplane(a,b)
float a,b;
{float x,y;
circle(x0,y0,R);
putpixel(x0,y0,RED);a=90-a;
if(b==0)circle(x0,y0,R);
else
if(b==90)
{x=x0+R*cos((a-90)/K);
y=y0-R*sin((a-90)/K);
line(x,y,2*x0-x,2*y0-y);
}
else
{x=x0-R*cos(a/K)*tan(b/K);
y=y0+R*sin(a/K)*tan(b/K);
arc(x,y,a-(90-b),a+90-b,R/cos(b/K));
}
return;
}
void drawline(a,b)
float a,b;

```

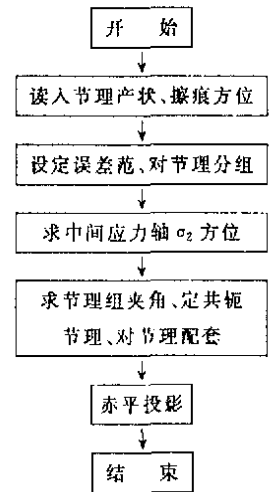


图 2 程序流程图

```

{float x,y;
a=90-a;
if(b==0)
{x=x0+R*cos(a/K);y=y0-R*sin(a/K);
line(x,y,2*x0-x,2*y0-y);
}
else if(b==90)putpixel(x0,y0,RED);
else
{x=x0+R*tan((45-b/2)/K)*cos(a/K);
y=y0-R*tan((45-b/2)/K)*sin(a/K);
line(x0,y0,x,y);
}
return;
}

```

```

float f1_b(b,c)
float b,c;
{float b1;
b1=asin(sin(b/K)*sin(c/K))*K;
return(b1);
}

```

```

float f1_a(a,b,c,d)
float a,b,c;
int d;
{float a1;
if(c!=90)
{if(d==1)
{a1=a+90-K*atan(tan(c/K)*cos(b/K));
if(a1>=360)a1=a1-360;
else a1=a1;
}
else
{a1=a-90+K*atan(tan(c/K)*cos(b/K));
if(a1<0)a1=a1+360;
else a1=a1;
}
}
else a1=a;
return(a1);
}

```

```
float f2(a1,b1,a2,b2)/ * 面面夹角 * /
```

```

float a1,b1,a2,b2;
{float a;
a=K*cos(acos(sin(b1/K)*sin(b2/K)*cos((a1-a2)/K)+cos
(b1/K)*cos(b2/K)));
return(a);
}

```

```

float f3(a1,b1,a2,b2)/ * 线线夹角 * /
float a2,b2,a1,b1;
{float b;
b=K*cos(acos(cos(b1/K)*cos(b2/K)*cos((a1-a2)/K)+sin
(b1/K)*sin(b2/K)));
return(b);
}

```

```

float f4_a_1(a1,b1,a2,b2)
{float x,y,z,a;
a1=90-a1; a2=90-a2;
x=-sin(b1/K)*cos(b2/K)*sin(a1/K)+sin(b2/K)*cos(b1/K)*sin(a2/K);
y=sin(b1/K)*cos(b2/K)*cos(a1/K)-sin(b2/K)*cos(b1/K)*cos(a2/K);
z=sin(b1/K)*sin(b2/K)*sin((a1-a2)/K);
if(z==0)
{if(x!=0&&y!=0)a=90-atan(y/x)*K;
else if(y==0&&x!=0)a=0;
else if(x==0&&y!=0)a=90;
else a=-1111;
}
else
{z=fabs(z);
if(x>0&&y!=0)a=90-atan(y/x)*K;
else if(x<0&&y!=0)a=90-atan(y/x)*K+180;
else if(y==0&&x>0)a=90;
else if(y==0&&x<0)a=270;
else if(x==0&&y>0)a=0;
else if(x==0&&y<0)a=180;
else a=-1000;
}
return(a);
}

```

```

}
float f4_a(a1,b1,a2,b2)
float a1,b1,a2,b2;
{float a;
a=f4_a_1(a1,b1,a2,b2);
if ((a1>=0&& a1<=180)&& (a2>a1&& a2<(a1
+180)))
a=180+a;
else
if(((a1>180&& a1<=360)&& (a2>a1))||((a1>
180&& a1<=360)&& (a2<(a1-180))))
a=180+a;
else a=a;
return(a);
}

float f4_b(a1,b1,a2,b2)
float a1,b1,a2,b2;
{float x,y,z,r1,b;
a1=90-a1;a2=90-a2;
x=-sin(b1/K)*cos(b2/K)*sin(a1/K)+sin
(b2/K)
*cos(b1/K)*sin(a2/K);
y=sin(b1/K)*cos(b2/K)*cos(a1/K)-
sin(b2/K)*cos(b1/K)*cos(a2/K);
z=sin(b1/K)*sin(b2/K)*sin((a1-a2)/K);
r1=sqrt(x*x+y*y+z*z);
if(z==0)
{if(x==0&&y==0)b=0;
else b=0;
}
else
{z=fabs(z);
if(x==0&&y==0) b=90;
else b=asin(z/r1)*K;
}
return(b);
}

void wf()
{circle(x0,y0,R);
line(x0-R,y0,x0+R,y0);
line(x0,y0-R,x0,y0+R);
gotoxy(38,6);printf("N");
gotoxy(38,18);printf("S");
gotoxy(52,12); printf("E");
gotoxy(25,12); printf("W");
}

main()
{int gdriver=9,gmode=2;
float maxa[20],maxb[20],sjla[20],sjlb[20];
float f_2[20][20],f_3[20][20],f4a[20][20],
float mint[20],minp[20],f4b[20][20];
int i,j,m,n,h=1,t[20],p[20],d[20],ts;
float a[20],b[20],c[20];
char ch;int dh;
initgraph(&gdriver,&gmode,"")(3)(4);
setlinestyle(0,0,1);
textmode(3);clrscr();
do
{gotoxy(26,1);printf("利用擦痕对节理配套微程
序");
gotoxy(26,2);printf("程序设计 张明华 王增
科");
gotoxy(30,3);printf("2000年5月");
gotoxy(1,6);printf("请输入数据:");
gotoxy(1,7);printf("点号:");scanf("%d",&dh);
gotoxy(1,8);printf("带擦痕节理组数:");
scanf("%d",&m);
for(i=0;i<m;i++)
{gotoxy(1,9);printf("第%d组带擦痕节理:",i+
1);
gotoxy(1,10);printf("节理产状:(a/b)");
scanf("%f/%f",&a[i],&b[i]);
gotoxy(1,11);printf("擦痕侧伏角:");
scanf("%f",&c[i]);
gotoxy(h,12);printf("擦痕侧伏向:(1|}-1)");
scanf("%d",&d[i]);
sjla[i]=f1_a(a[i],b[i],90-c[i],-d[i]);
sjlb[i]=f1_b(b[i],90-c[i]);
for(j=9;j<=12;j++){gotoxy(1,j);clreol();}
}
for(j=9;j<=12;j++){gotoxy(1,j);clreol();}
gotoxy(1,9);printf("节理组数:");
}

```

```

scanf("%d",&n);
for(i=0;i<n;i++)
{gotoxy(1,10+i);printf("第%d组节理产状:",i+
1);
scanf("%f/%f",&maxa[i],&maxb[i]);
}
/*求带擦痕节理和优势方位夹角*/
for(j=0;j<m;j++)
for(i=0;i<n;i++)
f_2[j][i]=f2(a[j],b[j],maxa[i],maxb[j]);
/*求最小夹角 mint[j]和所对应的优势方位地址 t
[j]*/
for(j=0;j<m;j++)
{mint[j]=f_2[j][0];t[j]=0;
for(i=1;i<n;i++)
{if(mint[j]>f_2[j][i])
{mint[j]=f_2[j][i];t[j]=i;}
else continue;
}
}
/*求于擦痕节理同组的优势方位跟其他优势方位
的公共轴
f4a[j][i]/f4b[j][i]*/
/*其中当 t[j]=i 即同一个的优势方位时应当跳过
去,这时
f4a[j][i]=0,f4b[j][i]=0*/
for(j=0;j<m;j++)
for(i=0;i<n;i++)
{f4a[j][i]=f4_a(maxa[t[j]],maxb[t[j]],maxa
[i],m
axb[i]);
f4b[j][i]=f4_b(maxa[t[j]],maxb[t[j]],maxa
[i],ma
xb[i]);}
/*求带擦痕节理所得的公共轴于上段公共轴夹角 f
_3[j][i]*/
for(j=0;j<m;j++)
for(i=0;i<n;i++)
{f_3[j][i]=f3(sjla[j],sjlb[j],f4a[j][i],f4b[j][
i]);}
/*最终判断出最小的公共轴夹角 minp[j],并求的
被配套的优势方位地址 p[j]值*/
for(j=0;j<m;j++)
{if(t[j]!=0)
{minp[j]=f_3[j][0];p[j]=0;
for(i=1;i<n;i++)
{if(t[j]==i)
{if(minp[j]>f_3[j][i])
{minp[j]=f_3[j][i];p[j]=i;}
else continue;}
else continue;}
}
else
{minp[j]=f_3[j][1];p[j]=1;
for(i=2;i<n;i++)
{if(minp[j]>f_3[j][i])
{minp[j]=f_3[j][i];p[j]=i;}
else continue;
} } }
wf();
/*最后求配套节理组*/
gotoxy(58,6);printf("No:%d",dh);
gotoxy(58,7);printf("节理产状:");
for(i=0;i<n;i++)
{gotoxy(65,8+i);printf("%d,%-5.2f/%-5.2f",
i+1,maxa[i],maxb[i]);}
gotoxy(65,8+n);printf("配套节理组:");
ts=0;
for(j=0;j<m;j++)
{if(mint[j]<=10&&minp[j]<=10)
{ts=ts+1;setcolor(RED);
drawplane(maxa[t[j]],maxb[t[j]]);
drawplane(maxa[p[j]],maxb[p[j]]);
gotoxy(56,8+n+ts);
printf("%-5.2f/%-5.2f",maxa[t[j]],maxb[t
[j]]);
gotoxy(56,8+n+1+ts);
printf("%-5.2f/%-5.2f",maxa[p[j]],maxb[p
[j]]);
}
else continue;
}
for(j=0;j<m;j++)
{drawline(sjla[j],sjlb[j]);
drawplane(a[j],b[j]);}

```

```

gotoxy(30,20);printf("节理配套赤平投影图");
gotoxy(32,23);printf("继续吗? ... (Y/N)");
ch=getch();cleardevice();
for(i=0;i<n;i++)
{maxa[i]=0;maxb[i]=0;}
for(j=0;j<m;j++)
{mint[j]=0;minp[j]=0;t[j]=0;p[j]=0;a[i]=0;
b[i]=0;c[i]=0;d[i]=0;sjla[i]=0;sjlb[i]=0;}
for(j=0;j<m;j++)
for(i=0;i<n;i++)
{f_2[j][i]=0;f_3[j][i]=0;}
}while(ch!='89'|ch=='121');
getch();closegraph();
}

```

该程序在 Turbo C 下编译、连接后即可运行,在 586 以上微机上运行通过;在 DOS 或 WINDOWS95/98/2000 的 MS-DOS 模式下运行,运行前需加载中文汉字系统(如 UC DOS 等),运行后按提示输入点号、带擦痕的节理组数及每组节理的产状、擦痕侧伏角和侧伏向、节理组数及每组节理的产状。其运行结果如图 3。

4 结语

利用擦痕对节理进行配套简单易行,可作为节理分期配套研究的辅助方法之一,同时,利用计算机对节理配套数据进行自动处理可大大提高工作效率。实际上,计算机在构造解析中的应用十分广泛,而且,通过设计计算机程序或软件进行地质构造定量研究及模拟已是构造地质学的发展趋势之一。

利用擦痕对节理配套微机程序
程序设计 张明华 王增科

2000.3.15

输入数据:

点号:12

带擦痕节理组数:2

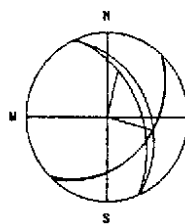
节理组数:4

第1组节理产状:142/34

第2组节理产状:134/31

第3组节理产状:56/43

第4组节理产状:65/34



节理配套赤平投影图

继续吗? ... (Y/N)

No:12

节理产状:

1. 142.00/34.00

2. 134.00/31.00

3. 56.00/43.00

4. 65.00/34.00

配套节理组:

134.00/31.00

65.00/34.00

图3 程序运行结果

参考文献

- 1 徐开礼,朱志澄. 构造地质学. 北京:地质出版社,1990.112~113.
- 2 汤经武,杨学敏等. 微型计算机在地质构造解析中的应用. 武汉:中国地质大学出版社,1989.5~26,202~204.
- 3 林学焦,刘力,晓夏. Turbo C 2.0 用户手册. 北京:学苑出版社,1993.126~138.
- 4 叶欣,王枫. Turbo C 2.0 参考手册. 北京:学苑出版社,1993.104~107.

(责任编辑:邓大玉)